

DEVICE FOR COPYING PROGRAM PLAN AND SOURCE PROGRAM

Publication number: JP6195214 (A)

Publication date: 1994-07-15

Inventor(s): UCHIKANEZAKI SEIICHI

Applicant(s): TOSHIBA CORP

Classification:

- international: G06F9/06; G06F9/06; (IPC1-7): G06F9/06

- European:

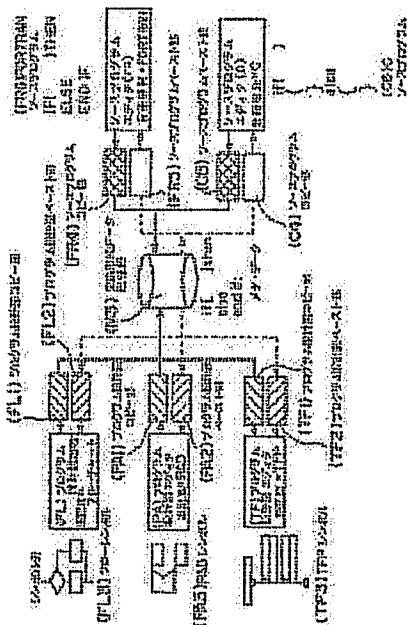
Application number: JP19920345040 19921225

Priority number(s): JP19920345040 19921225

Abstract of JP 6195214 (A)

PURPOSE: To improve productivity caused by diversion in an editor editing operation level and to easily copy the plan and the program of different notations and language classifications by providing an exchange system data storage means, program plan copying means/pasting means, and source program copying means/pasting means.

CONSTITUTION: An exchange system data storage part M3 preserves exchange system data showing the algorithm of the program without depending on the notations and the type of a programming language. The program plan copying part FL1 converts a flow chart symbol into exchange system data, and the program plan pasting part FL2 converts the algorithm shown as exchange system data into the flow chart symbol and pastes it on a flow chart editor FL. The source program copying part FR4 and the pasting part FR5 converts the FORTRAN source of a source program into exchange system data, for example.



Data supplied from the esp@cenet database — Worldwide

(19)日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開平6-195214

(43)公開日 平成6年(1994)7月15日

(51)Int.Cl.⁵
G 0 6 F 9/06

識別記号 430 G 庁内整理番号 9367-5B

FI

技術表示箇所

審査請求 未請求 請求項の数 5 (全 8 頁)

(21)出願番号 特願平4-345040

(22)出願日 平成4年(1992)12月25日

(71)出願人 000003078

株式会社東芝

神奈川県川崎市幸区堀川町72番地

(72)発明者 内金崎 誠一

東京都府中市東芝町1番地 株式会社東芝
府中工場内

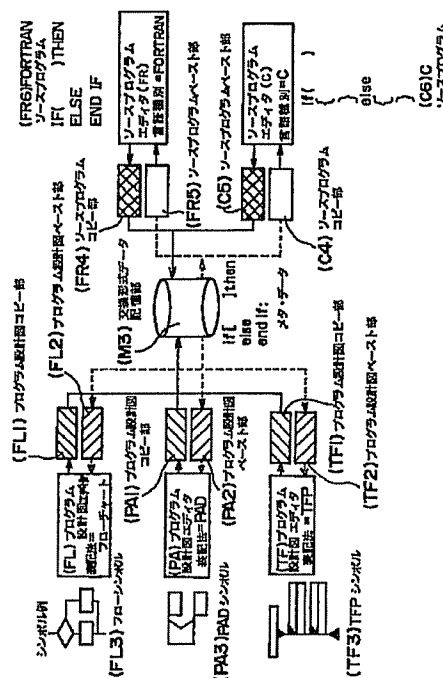
(74)代理人 弁理士 鈴江 武彦

(54)【発明の名称】 プログラム設計図およびソースプログラムの複写装置

(57) 【要約】

【目的】本発明は、プログラム設計図およびソースプログラムのエディタ編集操作レベルでの流用による生産性向上を図り、異なる表記法、言語種別のプログラム設計図、ソースプログラムの複写を簡単に行えることを主要な目的とする。

【構成】本発明は、プログラム設計図の表記法とプログラミング言語の種類に依存しない形式の交換形式データを保存する交換形式データ記憶手段、プログラム設計図エディタで選択されたシンボルを交換形式データに変換するプログラム設計図シンボルコピー手段、交換形式データを指定された表記法のプログラム設計図に変換し入力するプログラム設計図シンボルペースト手段、ソースプログラム構文エディタで選択されたソースプログラムを交換形式データに変換するソースプログラムコピー手段、交換形式データを指定されたプログラミング言語の種類に変換し入力するソースプログラムペースト手段とを備えたことを特徴としている。



【特許請求の範囲】

【請求項1】 プログラム設計図の表記法とプログラミング言語の種類に依存しない形式の交換形式データを保存するための交換形式データ記憶手段と、プログラム設計図エディタにおいて選択されたシンボルを交換形式データに変換し、前記交換形式データ記憶手段に保存するプログラム設計図シンボルコピー手段と、前記交換形式データ記憶手段に保存されている交換形式データを指定された表記法のプログラム設計図に変換し、前記プログラム設計図エディタに入力するプログラム設計図シンボルペースト手段と、ソースプログラム構文エディタにおいて選択されたソースプログラムを交換形式データに変換し、前記交換形式データ記憶手段に保存するソースプログラムコピー手段と、前記交換形式データ記憶手段に保存されている交換形式データを指定されたプログラミング言語の種類に変換し、前記ソースプログラム構文エディタに入力するソースプログラムペースト手段と、を備えて成ることを特徴とするプログラム設計図およびソースプログラムの複写装置。

【請求項2】 前記プログラム設計図エディタからプログラム設計図エディタへのプログラム設計図シンボルの複写（コピー＋ペースト）を行なう場合に、コピーを行なう側のプログラム設計図エディタのプログラム設計図シンボルコピー手段で、選択されたシンボルをプログラム設計図の表記法に依存しない形式の交換形式データに変更し、ペーストを行なう側のプログラム設計図エディタのプログラム設計図シンボルペースト手段で、交換形式データに変換された選択シンボル（被複写シンボル）を指定された表記法のプログラム設計図シンボルに変換し入力するようにしたことを特徴とする請求項1に記載のプログラム設計図およびソースプログラムの複写装置。

【請求項3】 前記プログラム設計図エディタからソースプログラム構文エディタへのプログラム設計図シンボルの複写（コピー＋ペースト）を行なう場合に、プログラム設計図エディタのプログラム設計図シンボルコピー手段で、選択されたプログラム設計図シンボルをプログラム設計図の表記法とプログラミング言語の種類に依存しない形式の交換形式データに変更し、ソースプログラム構文エディタのソースプログラムペースト手段で、交換形式データに変換された選択シンボル（被複写シンボル）を指定された種類のプログラミング言語に変換し入力するようにしたことを特徴とする請求項1に記載のプログラム設計図およびソースプログラムの複写装置。

【請求項4】 前記ソースプログラム構文エディタからプログラム設計図エディタへのソースプログラムの複写（コピー＋ペースト）を行なう場合に、ソースプログラム構文エディタのソースプログラムコピー手段で、選択

されたソースプログラムをプログラム設計図の表記法とプログラミング言語の種類に依存しない形式の交換形式データに変更し、プログラム設計図エディタのプログラム設計図シンボルペースト手段で、交換形式データに変換された選択ソースプログラム（被複写ソースプログラム）を指定された表記法のシンボルに変換し入力するようにしたことを特徴とする請求項1に記載のプログラム設計図およびソースプログラムの複写装置。

【請求項5】 前記ソースプログラム構文エディタからプログラム設計図エディタへのソースプログラムの複写（コピー＋ペースト）を行なう場合に、複写元のソースプログラム構文エディタのソースプログラムコピー手段で、選択されたソースプログラムをプログラム設計図の表記法とプログラミング言語の種類に依存しない形式の交換形式データに変更し、複写先のプログラム設計図エディタのプログラム設計図シンボルペースト手段で、交換形式データに変換された選択ソースプログラム（被複写ソースプログラム）を指定された言語種別のソースプログラムに変換し入力するようにしたことを特徴とする請求項1に記載のプログラム設計図およびソースプログラムの複写装置。

【発明の詳細な説明】

【0001】

【産業上の利用分野】本発明は、プログラム設計図およびソースプログラムの複写を行なう装置に係り、特に既存のプログラム設計図およびソースプログラムの、エディタ編集操作レベルでの流用による生産性向上を図りつつ、異なる表記法のプログラム設計図エディタ間、異なる言語種別のソースプログラム構文エディタ間、およびプログラム設計図エディタ、ソースプログラム構文エディタ間での、プログラム設計図およびソースプログラムの複写を簡単に行ない得るようにしたプログラム設計図およびソースプログラムの複写装置に関するものである。

【0002】

【従来の技術】従来、プログラム設計図エディタおよびソースプログラムエディタ間では、同一計算機上で動作しても、その設計図シンボル、テキスト等は複写が行なえず、設計図シンボル、テキスト等の流用による、エディタ編集操作レベルでの生産性向上が期待できなかった。

【0003】一方、近年では、計算機の機能が向上し、プログラム設計図エディタ、ソースプログラムエディタとも、同一計算機上で動作し、同一の表記法、言語種別であれば、その設計図シンボル、テキストは簡単に複写・流用できるようになり、生産性向上が図られてきている。

【0004】しかしながら、現在では、取り扱うアプリケーションの性格が多種多様となり、設計手法も多数提唱され、これらを背景に、プログラム設計図の作成に使

用される表記法やソースプログラムの使用言語種別も、同様に多種多用となってきた。このような状況の中、同一の表記法、言語種別でしか、設計図シンボルおよびテキストの複写が行えないのでは、エディタ編集レベルでの生産性向上を期待することができない。

【0005】

【発明が解決しようとする課題】以上のように、従来のプログラム設計図およびソースプログラムの複写方法においては、エディタ編集レベルでの生産性向上を期待ができないという問題があった。

【0006】本発明の目的は、異なる表記法のプログラム設計図エディタ間で、論理的アルゴリズムは同一のままプログラム設計図の表記法のみを変更して設計図シンボルおよびテキストを複写することができ、またプログラム設計図エディタとソースプログラム構文エディタ間で、論理的アルゴリズムは同一のまま抽象度レベル（設計図レベル、コードレベル）のみを変更してアルゴリズムを複写することができ、さらに異なる言語種別のソースプログラム構文エディタ間で、論理的アルゴリズムは同一のまま言語種別を自動変換してソースプログラムを複写することが可能な極めて信頼性の高いプログラム設計図およびソースプログラムの複写装置を提供することにある。

【0007】

【課題を解決するための手段】上記の目的を達成するために本発明では、プログラム設計図の表記法とプログラミング言語の種類に依存しない形式の交換形式データを保存するための交換形式データ記憶手段と、プログラム設計図エディタにおいて選択されたシンボルを交換形式データに変換し、交換形式データ記憶手段に保存するプログラム設計図シンボルコピー手段と、交換形式データ記憶手段に保存されている交換形式データを指定された表記法のプログラム設計図に変換し、プログラム設計図エディタに入力するプログラム設計図シンボルペースト手段と、ソースプログラム構文エディタにおいて選択されたソースプログラムを交換形式データに変換し、交換形式データ記憶手段に保存するソースプログラムコピー手段と、交換形式データ記憶手段に保存されている交換形式データを指定されたプログラミング言語の種類に変換し、ソースプログラム構文エディタに入力するソースプログラムペースト手段とを備えて構成している。

【0008】ここで、特に上記プログラム設計図エディタからプログラム設計図エディタへのプログラム設計図シンボルの複写（コピー＋ペースト）を行なう場合に、コピーを行なう側のプログラム設計図エディタのプログラム設計図シンボルコピー手段で、選択されたシンボルをプログラム設計図の表記法に依存しない形式の交換形式データに変更し、ペーストを行なう側のプログラム設計図エディタのプログラム設計図シンボルペースト手段で、交換形式データに変換された選択シンボル（被複写

シンボル）を指定された表記法のプログラム設計図シンボルに変換し入力するようにしている。

【0009】また、上記プログラム設計図エディタからソースプログラム構文エディタへのプログラム設計図シンボルの複写（コピー＋ペースト）を行なう場合に、プログラム設計図エディタのプログラム設計図シンボルコピー手段で、選択されたプログラム設計図シンボルをプログラム設計図の表記法とプログラミング言語の種類に依存しない形式の交換形式データに変更し、ソースプログラム構文エディタのソースプログラムペースト手段で、交換形式データに変換された選択シンボル（被複写シンボル）を指定された種類のプログラミング言語に変換し入力するようにしている。

【0010】さらに、上記ソースプログラム構文エディタからプログラム設計図エディタへのソースプログラムの複写（コピー＋ペースト）を行なう場合に、ソースプログラム構文エディタのソースプログラムコピー手段で、選択されたソースプログラムをプログラム設計図の表記法とプログラミング言語の種類に依存しない形式の交換形式データに変更し、プログラム設計図エディタのプログラム設計図シンボルペースト手段で、交換形式データに変換された選択ソースプログラム（被複写ソースプログラム）を指定された表記法のシンボルに変換し入力するようにしている。

【0011】さらにまた、上記ソースプログラム構文エディタからプログラム設計図エディタへのソースプログラムの複写（コピー＋ペースト）を行なう場合に、複写元のソースプログラム構文エディタのソースプログラムコピー手段で、選択されたソースプログラムをプログラム設計図の表記法とプログラミング言語の種類に依存しない形式の交換形式データに変更し、複写先のプログラム設計図エディタのプログラム設計図シンボルペースト手段で、交換形式データに変換された選択ソースプログラム（被複写ソースプログラム）を指定された言語種別のソースプログラムに変換し入力するようにしている。

【0012】

【作用】従って、本発明のプログラム設計図およびソースプログラムの複写装置においては、以上のような手段を備えたことにより、例えばある表記法で作成中のプログラム設計図のある任意の一部分を、他の表記法のプログラム設計図、あるいはソースプログラム構文エディタに複写・流用したい場合には、以下のような操作により、簡単に複写・流用を行なうことができる。

【0013】まず、複写・流用元のプログラム設計図を編集するプログラム設計図エディタで、流用したいアルゴリズムを表わすシンボルを選択し、プログラム設計図シンボルコピー手段を動作させる。これにより、複写・流用対象として選択したアルゴリズムは、表記法、言語種別に依存しない交換形式データに変換される。この状態で、複写・流用先のプログラム設計図エディタ、また

はソースプログラム構文エディタで、それぞれプログラム設計図シンボルペースト手段、ソースプログラムペースト手段を動作させることにより、表記法、言語種別に依存しない交換形式データが、複写・流用先のエディタに対応する表記法のプログラム設計図、言語種別のソースプログラムに変換されて、複写・流用が行なわれる。

【0014】さらに、以上のような手段を備えたことにより、ある言語種別で作成されたソースプログラムのある任意のアルゴリズムを、他の言語種別のソースプログラム構文エディタ、あるいは他の表記法のプログラム設計図に複写・流用したい場合には、以下のような操作により、簡単に複写・流用を行なうことができる。

【0015】まず、複写・流用元のソースプログラム構文エディタで、流用したいアルゴリズムを表わすソースプログラムを選択し、ソースプログラムコピー手段を動作させる。これにより、複写・流用対象として選択したアルゴリズムは、表記法、言語種別に依存しない交換形式データに変換される。この状態で、複写・流用先のソースプログラム構文エディタ、プログラム設計図エディタで、それぞれソースプログラムペースト手段、プログラム設計図シンボルペースト手段を動作させることにより、表記法、言語種別に依存しない交換形式データが、複写・流用先のエディタに対応する表記法のプログラム設計図、言語種別のソースプログラムに変換されて、複写・流用が行なわれる。

【0016】

【実施例】以下、本発明の一実施例について図面を参照して詳細に説明する。図1は、本発明によるプログラム設計図およびソースプログラムの複写装置の全体構成例を示す機能ブロック図である。

【0017】すなわち、本実施例のプログラム設計図およびソースプログラムの複写装置は、図1に示すように、交換形式データ記憶部M3と、プログラム設計図エディタFLと、プログラム設計図コピー部FL1と、プログラム設計図ペースト部FL2と、プログラム設計図エディタPAと、プログラム設計図コピー部PA1と、プログラム設計図ペースト部PA2と、プログラム設計図エディタTFと、プログラム設計図コピー部TF1と、プログラム設計図ペースト部TF2と、ソースプログラム構文エディタFRと、ソースプログラムコピー部FR4と、ソースプログラムペースト部FR5と、ソースプログラム構文エディタCと、ソースプログラムコピー部C4と、ソースプログラムペースト部C5とから構成している。

【0018】ここで、交換形式データ記憶部M3は、プログラム設計図の表記法、プログラミング言語の種類に依存しない、プログラムのアルゴリズムを表わす交換形式データを保存するためのものである。また、プログラム設計図エディタFLは、フローチャートを作成・編集するものである。

【0019】さらに、プログラム設計図コピー部FL1は、プログラム設計図エディタFL上で選択されたフローチャートシンボルを交換形式データに変換し、交換形式データ記憶部M3に保存するものである。

【0020】さらにまた、プログラム設計図ペースト部FL2は、交換形式データとして表わされるアルゴリズムをフローチャートシンボルに変換し、フローチャートエディタFLにペーストするものである。一方、プログラム設計図エディタPAは、PADを作成・編集するものである。

【0021】また、プログラム設計図コピー部PA1は、プログラム設計図エディタPA上で選択されたPADシンボルを交換形式データに変換し、交換形式データ記憶部M3に保存するものである。

【0022】さらに、プログラム設計図ペースト部PA2は、交換形式データとして表わされるアルゴリズムをPADシンボルに変換し、PADシンボルにペーストするものである。一方、プログラム設計図エディタTFは、TFPシンボルを作成・編集するものである。

【0023】また、プログラム設計図コピー部TF1は、プログラム設計図エディタTF上で選択されたTFPシンボルを交換形式データに変換し、交換形式データ記憶部M3に保存するものである。

【0024】さらに、プログラム設計図ペースト部TF2は、交換形式データとして表わされるアルゴリズムをTFPシンボルに変換し、TFPシンボルにペーストするものである。一方、ソースプログラム構文エディタFRは、FORTRAN言語の編集を行なうものである。

【0025】また、ソースプログラムコピー部FR4は、ソースプログラム構文エディタFR上で選択されたFORTRANソースを交換形式データに変換し、交換形式データ記憶部M3に保存するものである。

【0026】さらに、ソースプログラムペースト部FR5は、交換形式データとして表わされるアルゴリズムをFORTRANソースに変換し、FORTRANソース中にペーストするものである。一方、ソースプログラム構文エディタCは、C言語の編集を行なうものである。

【0027】また、ソースプログラムコピー部C4は、ソースプログラム構文エディタC上で選択されたCソースを交換形式データに変換し、交換形式データ記憶部M3に保存するものである。さらに、ソースプログラムペースト部C5は、交換形式データとして表わされるアルゴリズムをCソースに変換し、Cソース中にペーストするものである。次に、以上のように構成した本実施例のプログラム設計図およびソースプログラムの複写装置の作用について説明する。

(複写・流用処理：その1)

【0028】最初に、異なる表記法間でのプログラム設計図の複写・流用処理の概要を、プログラム設計図エディタFLからフローチャートシンボルFL3をコピー

し、プログラム設計図エディタPAにPADシンボルPA3としてペーストする場合を例として述べる。

【0029】まず、プログラム設計図エディタFL上で複写・流用したいフローチャートシンボルFL3を選択し、プログラム設計図コピー部FL1を動作させる。すると、プログラム設計図コピー部FL1は、選択されて

```

1. if [ ] then <--+----- 「分岐処理」
   2. □ ; <--+ |
   else      |<--+
   3. □ ; <--+----- 「順次処理」
end if ; <--+

```

【0031】このようにして、プログラム設計図エディタFL上のフローチャートシンボルFL3で示されるアルゴリズムが、交換形式データとして表わされたら、次に複写・流用先のプログラム設計図エディタPA上で、プログラム設計図ペースト部PA2を動作させる。すると、プログラム設計図ペースト部PA2は、交換形式データPAを入力し、交換形式データで示される分岐処理、順次処理、処理の順序を、PADシンボルPA3のそれぞれの構成要素図形に変換し、プログラム設計図エディタPA上のプログラム設計図にペーストする。

【0032】これにより、プログラム設計図エディタFL上で選択したフローチャートシンボルFL3が、プログラム設計図エディタPA上へPADシンボルPA3として複写・流用される。

【0033】すなわち、異なる表記法のプログラム設計図エディタ間で、論理的アルゴリズムは同一のまま、プログラム設計図の表記法のみを変更して、プログラム設計図シンボルおよびテキストを複写することができる。

【0034】なお、上記と同様の処理手順で、プログラム設計図エディタFL上で選択したフローチャートシンボルFL3を、プログラム設計図エディタTF上へTEPシンボルTF3として複写・流用できる。

(複写・流用処理：その2)

【0035】次に、プログラム設計図エディタ上で選択したプログラム設計図シンボルの、ソースプログラム構文エディタへの複写・流用処理の概要を、プログラム設計図エディタFLからフローチャートシンボルFL3をコピーし、プログラム構文エディタFRへFORTRANソースFR6としてペーストする場合を例として述べる。

【0036】まず、プログラム設計図エディタFL上で複写・流用したいフローチャートシンボルを選択し、プログラム設計図コピー部FL1を動作させる。すると、プログラム設計図コピー部FL1は、選択されているフローチャートシンボルFL3を構成する図形要素を、菱形図形は分岐処理、四角図形は順次処理、線は処理の順序として認識し、これら分岐処理、順次処理、処理の順序を、プログラム設計図の表記法に依存しない、前述したような交換形式データに変換する。

いるフローチャートシンボルFL3を構成する図形要素を、菱形図形は分岐処理、四角図形は順次処理、線は処理の順序として認識し、これら分岐処理、順次処理、処理の順序を、プログラム設計図の表記法に依存しない、例えば以下に示すような交換形式データに変換する。

【0030】

【0037】このようにして、プログラム設計図エディタFL上のフローチャートシンボルFL3で示されるアルゴリズムが、交換形式データとして表わされたら、次に複写・流用先のソースプログラム構文エディタFR上で、ソースプログラムペースト部FR5を動作させる。すると、ソースプログラムペースト部FR5は、交換形式データPAを入力し、交換形式データで示される分岐処理、順次処理、処理の順序を、FORTRAN言語FR6のそれぞれの構文に変換し、ソースプログラム構文エディタFR上のソースプログラムにペーストする。

【0038】これにより、プログラム設計図エディタFL上で選択したフローチャートシンボルFL3が、ソースプログラム構文エディタFR上へFORTRANソースプログラムFR6として複写・流用される。

【0039】すなわち、プログラム設計図エディタとソースプログラム構文エディタ間で、論理的アルゴリズムは同一のまま、アルゴリズム表現の抽象度レベルのみを変更(設計図レベルからコードレベルへ)して、アルゴリズムを複写することができる。

【0040】なお、上記と同様の処理手順で、プログラム設計図エディタFL上で選択したフローチャートシンボルFL3を、ソースプログラム構文エディタC上へCソースプログラムC6として複写・流用できる。

(複写・流用処理：その3)

【0041】次に、プログラム構文エディタ上で選択したソースプログラムの、プログラム設計図エディタへのプログラム設計図シンボルとしての複写・流用処理の概要を、ソースプログラム構文エディタFR上のFORTRANソースプログラムFR6をコピーし、プログラム設計図エディタFLへフローチャートシンボルFL3としてペーストする場合を例として述べる。

【0042】まず、プログラム構文エディタFR上で複写・流用したいアルゴリズムのソースプログラムを選択し、ソースプログラムコピー部FR4を動作させる。すると、ソースプログラムコピー部FR4は、選択されているソースプログラムを構成する、順次処理、分岐処理等の基本制御構造を認識し、これらをプログラミング言語の種別に依存しない、前述したような交換形式データに変換する。

【0043】このようにして、プログラム構文エディタFR上で選択されたソースプログラムが交換形式データとして表わされたら、次に複写・流用先のプログラム設計図エディタFL上で、プログラム設計図ペースト部FL2を動作させる。すると、プログラム設計図ペースト部FL2は、交換形式データM3を入力し、交換形式データで示される分岐処理、順次処理、処理の順序を、フローチャートシンボルFL3のそれぞれの構成要素図形に変換し、プログラム設計図エディタFL上のプログラム設計図にペーストする。

【0044】これにより、ソースプログラム構文エディタFR上で選択したソースプログラムが、プログラム設計書エディタFL上へフローチャートシンボルFL3として複写・流用される。

【0045】すなわち、ソースプログラム構文エディタとプログラム設計図エディタ間で、論理的アルゴリズムは同一のまま、アルゴリズム表現の抽象度レベルのみを変更（コードレベルから設計図レベルへ）して、アルゴリズムを複写することができる。

【0046】なお、上記と同様の処理手順で、ソースプログラム構文エディタFR上で選択したソースプログラムを、プログラム設計図エディタPA上へPADシンボルとして、あるいはプログラム設計図エディタTF上へTFPシンボルとして複写・流用できる。

（複写・流用処理：その4）

【0047】次に、ある言語種別のプログラム構文エディタ上で選択したソースプログラムを、他の言語種別のプログラム構文エディタ上へ言語変換して複写・流用する処理の概要を、ソースプログラム構文エディタFR上のFORTRANソースプログラムFR6をコピーし、ソースプログラム構文エディタC上へCソースプログラムC6としてペーストする場合を例として述べる。

【0048】まず、プログラム構文エディタFR上で複写・流用したいアルゴリズムのソースプログラムを選択し、ソースプログラムコピー部FR4を動作させる。すると、ソースプログラムコピー部FR4は、選択されているソースプログラムを構成する、順次処理、分岐処理等の基本制御構造を認識し、これらをプログラミング言語の種別に依存しない、前述したような交換形式データに変換する。

【0049】このようにして、プログラム構文エディタFR上で選択されたソースプログラムが交換形式データとして表わされたら、次に複写・流用先のソースプログラム構文エディタC上で、ソースプログラムペースト部C5を動作させる。すると、ソースプログラムペースト部C5は、交換形式データM3を入力し、交換形式データで示される分岐処理、順次処理、処理の順序を、C言語C6のそれぞれの構文に変換し、ソースプログラム構文エディタC上のCソースプログラムにペーストする。

【0050】これにより、ある言語種別のプログラム構

文エディタ上で選択したソースプログラムが、他の言語種別のプログラム構文エディタ上へ言語変換して複写・流用される。

【0051】すなわち、異なる言語種別のソースプログラム構文エディタ間で、論理的アルゴリズムは同一のまま、言語種別を自動変換して、ソースプログラムを複写することができる。

【0052】上述したように、本実施例のプログラム設計図およびソースプログラムの複写装置は、プログラム設計図の表記法、プログラミング言語の種類に依存しない、プログラムのアルゴリズムを表わす交換形式データを保存するための交換形式データ記憶部M3と、フローチャートを作成・編集するプログラム設計図エディタFLと、プログラム設計図エディタFL上で選択されたフローチャートシンボルを交換形式データに変換し、交換形式データ記憶部M3に保存するプログラム設計図コピー部FL1と、交換形式データとして表わされるアルゴリズムをフローチャートシンボルに変換し、フローチャートエディタFLにペーストするプログラム設計図ペースト部FL2と、プログラム設計図エディタPAと、プログラム設計図エディタPA上で選択されたPADシンボルを交換形式データに変換し、交換形式データ記憶部M3に保存するプログラム設計図コピー部PA1と、交換形式データとして表わされるアルゴリズムをPADシンボルに変換し、PADシンボルにペーストするプログラム設計図ペースト部PA2と、TFPシンボルを作成・編集するプログラム設計図エディタTFと、プログラム設計図エディタTF上で選択されたTFPシンボルを交換形式データに変換し、交換形式データ記憶部M3に保存するプログラム設計図コピー部TF1と、交換形式データとして表わされるアルゴリズムをTFPシンボルに変換し、TFPシンボルにペーストするプログラム設計図ペースト部TF2と、FORTRAN言語の編集を行なうソースプログラム構文エディタFRと、ソースプログラム構文エディタFR上で選択されたFORTRANソースを交換形式データに変換し、交換形式データ記憶部M3に保存するソースプログラムコピー部FR4と、交換形式データとして表わされるアルゴリズムをFORTRANソースに変換し、FORTRANソース中にペーストするソースプログラムペースト部FR5と、C言語の編集を行なうソースプログラム構文エディタCと、ソースプログラム構文エディタC上で選択されたCソースを交換形式データに変換し、交換形式データ記憶部M3に保存するソースプログラムコピー部C4と、交換形式データとして表わされるアルゴリズムをCソースに変換し、Cソース中にペーストするソースプログラムペースト部C5とから構成したものである。

【0053】従って、異なる表記法のプログラム設計図エディタ間で、論理的アルゴリズムは同一のままプログラム設計図の表記法のみを変更して設計図シンボルおよ

びテキストを複写することができ、またプログラム設計図エディタとソースプログラム構文エディタ間で、論理的アルゴリズムは同一のまま抽象度レベル（設計図レベル、コードレベル）のみを変更してアルゴリズムを複写することができ、さらに異なる言語種別のソースプログラム構文エディタ間で、論理的アルゴリズムは同一のまま言語種別を自動変換してソースプログラムを複写することができる。

【0054】以上により、異なる表記法のプログラム設計図間、異なる言語種別のソースプログラム間、およびプログラム設計図とソースプログラム間で、アルゴリズム等の複写・流用を極めて簡単に行なうことができ、エディタ編集レベルでのアルゴリズム、ソースコードの流用促進、生産性向上を図ることが可能となる。

【0055】

【発明の効果】以上説明したように本発明によれば、プログラム設計図の表記法とプログラミング言語の種類に依存しない形式の交換形式データを保存するための交換形式データ記憶手段と、プログラム設計図エディタにおいて選択されたシンボルを交換形式データに変換し、交換形式データ記憶手段に保存するプログラム設計図シンボルコピー手段と、交換形式データ記憶手段に保存されている交換形式データを指定された表記法のプログラム設計図に変換し、プログラム設計図エディタに入力するプログラム設計図シンボルペースト手段と、ソースプログラム構文エディタにおいて選択されたソースプログラムを交換形式データに変換し、交換形式データ記憶手段に保存するソースプログラムコピー手段と、交換形式データ記憶手段に保存されている交換形式データを指定さ

れたプログラミング言語の種類に変換し、ソースプログラム構文エディタに入力するソースプログラムペースト手段とを備えて構成したので、異なる表記法のプログラム設計図エディタ間で、論理的アルゴリズムは同一のままプログラム設計図の表記法のみを変更して設計図シンボルおよびテキストを複写することができ、またプログラム設計図エディタとソースプログラム構文エディタ間で、論理的アルゴリズムは同一のまま抽象度レベル（設計図レベル、コードレベル）のみを変更してアルゴリズムを複写することができ、さらに異なる言語種別のソースプログラム構文エディタ間で、論理的アルゴリズムは同一のまま言語種別を自動変換してソースプログラムを複写することが可能な極めて信頼性の高いプログラム設計図およびソースプログラムの複写装置が提供できる。

【図面の簡単な説明】

【図1】本発明によるプログラム設計図およびソースプログラムの複写装置の一実施例を示す機能ブロック図。

【符号の説明】

M3…交換形式データ記憶部、FL…プログラム設計図エディタ、FL1…プログラム設計図コピー部、FL2…プログラム設計図ペースト部、PA…プログラム設計図エディタ、PA1…プログラム設計図コピー部、PA2…プログラム設計図ペースト部、TF…プログラム設計図エディタ、TF1…プログラム設計図コピー部、TF2…プログラム設計図ペースト部、FR…ソースプログラム構文エディタ、FR4…ソースプログラムコピー部、FR5…ソースプログラムペースト部、C…ソースプログラム構文エディタ、C4…ソースプログラムコピー部、C5…ソースプログラムペースト部。

【図1】

